

Nohai Alexandru

Grupa 234

## Proiect SGBD

# Administrarea agențiilor de turism

## Cerința 1

Proiectul are ca scop administrarea a mai multor agenții de turism, fiecare având angajați dar și clienți care își pot cumpăra oferta de vacanta dorita. In cadrul tabelelor pe care le-am creat si vor fi evidențiate ulterior in proiect, am realizat o structura care sa tina cont de orice poate oferi o oferta de vacanta (Beneficii, Transport, Tara, Oras, Cazare), dar sa si introducă istoricul de achiziționare a fiecărui comparator.

1. Un angajat aparține unui singur departament si unei singure agenții de turism
2. O agenție de turism poate avea mai multe oferte de vacanta
3. Un cumpărător poate achiziționa mai multe oferte de vacanta
4. In cazul in care un comparator nu vrea sa se folosească de o modalitate de transport o sa fie introdus in tabela o valoare de transport propriu
5. Prețul pentru o oferta de vacanta este același pentru un oraș dintr-o anumita tara indiferent de cazare
6. Prețul se specifică in tabela OFERTA\_IMPLICA\_TRANSPORT deci fluctuează in funcție de modalitatea de transport oferita
7. Un cumpărător achiziționează o oferta de vacanta cu o singura modalitate de transport, care va conta pentru preț

### Entități si chei primare

AGENTIE\_TURISM (#id\_agentie\_turism)

- Entitate care retine informații generale despre o agenție de turism

ANGAJAT (#id\_angajat)

- Entitate care retine informații despre angajat, cat si despre agenția si departamentul din care face parte

OFERTA\_VACANTA (#id\_oferta\_vacanta)

- Entitate care retine informații despre o oferta care aparține unei agenții de turism

BENEFICIU (#id\_beneficiu)

- Entitate care retine informații despre beneficiile care pot apărea intr-o oferta de vacanta

MOD\_TRANSPORT (#id\_mod\_transport)

- Entitate care retine informații despre modalitățile de deplasare pe care le pot conține ofertele de vacanta

TARA (#id\_tara)

- Entitate care retine informații despre numele tarii pentru o oferta de vacanta

ORAS (#id\_oras)

- Entitate care retine informații despre numele orașului care aparține unei țări

CAZARE (#id\_cazare)

- Entitate care retine informații despre numele cazării care aparține unui oraș

### **Relații si cardinalități**

AGENTIE\_TURISM – ANGAJAT

- O agenție de turism are mai mulți angajați, iar un angajat aparține unei singure agenții de turism
- Cardinalitate: 1 – M(1)

AGENTIE\_TURISM – OFERTA\_VACANTA

- O agenție de turism are mai multe oferte de vacanta, iar o oferta de vacanta specifica ca beneficia si preț aparține unei singure agenții de turism.
- Cardinalitate: 1 – M(1)

OFERTA\_VACANTA – CUMPARATOR

- O oferta de vacanta specifica poate fi achiziționata de mai mulți sau niciun comparator, iar un comparator poate sa fi achiziționat una sau mai multe oferte de vacanta
- Cardinalitate: M(1) – M(0)

OFERTA\_VACANTA – BENEFICIU

- O oferta de vacanta poate conține mai multe sau niciun beneficiu, iar un beneficiu poate fi prezent in mai multe sau nicio oferta
- Cardinalitate: M(0) – M(0)

OFERTA\_VACANTA – MOD\_TRANSPORT

- Pentru un tip de oferta de vacanta, cumpărătorul poate alege între unul sau mai multe moduri de transport, iar un mod de transport poate fi folosit pentru mai multe sau nicio oferta de vacanta
- Cardinalitate: M(0) – M(1)

OFERTA\_VACANTA – TARA

- O oferta de vacanta cuprinde o singura tara, iar o tara poate fi prezenta in mai multe sau nicio oferta de vacanta
- Cardinalitate: M(0) – 1

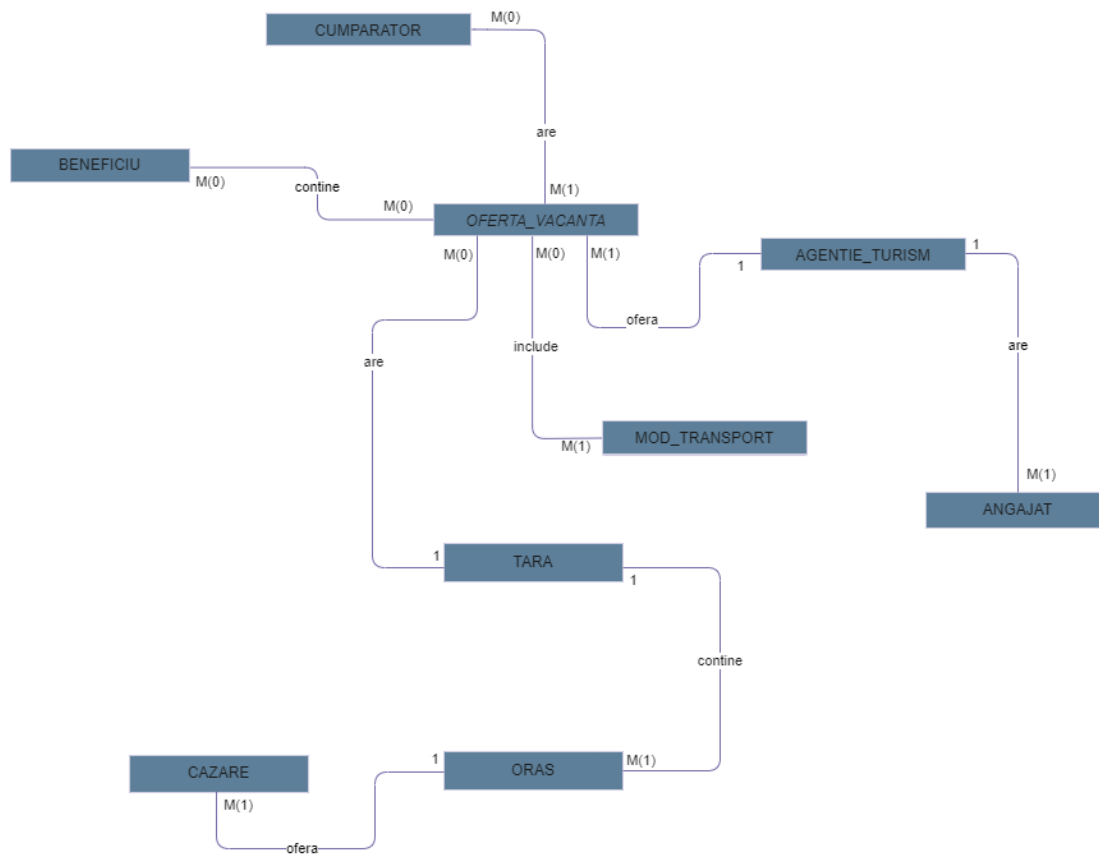
## TARA – ORAS

- O tara conține unul sau mai multe orașe, iar un oraș aparține unei singure tari
- Cardinalitate: 1 - M(1)

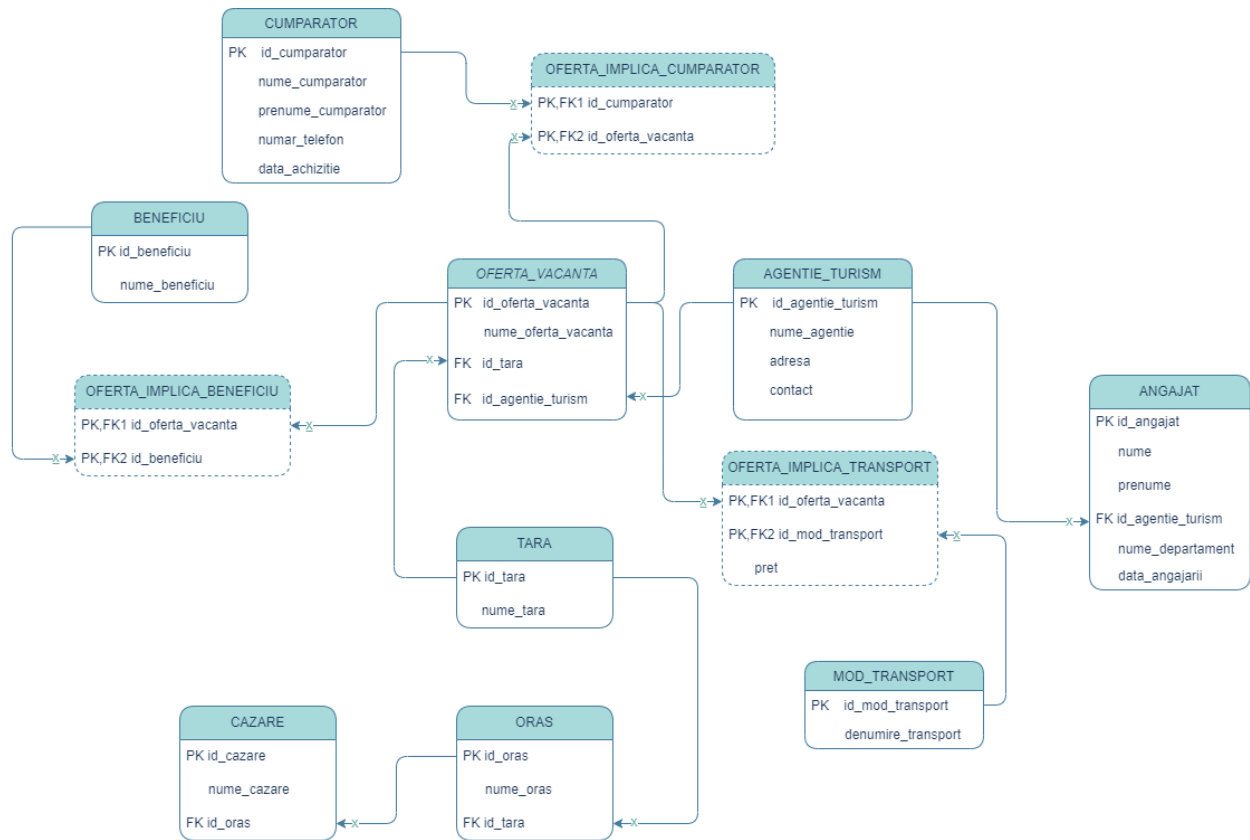
## ORAS – CAZARE

- Un oraș poate avea una sau mai multe cazări, iar o cazare se afla intr-un singur oraș
- Cardinalitate: 1 – M(1)

## Cerința 2 – Diagrama ERD



### Cerința 3 – Diagrama Conceptuala



### Cerința 4

```
CREATE TABLE AGENTIE_TURISM (
    id_agentie_turism INT PRIMARY KEY,
    nume_agentie VARCHAR2(25) NOT NULL,
    adresa VARCHAR2(100) NOT NULL,
    contact VARCHAR2(11) NOT NULL
);
```

```
CREATE TABLE ANGAJAT (  
    id_angajat INT PRIMARY KEY,  
    nume VARCHAR2(20) NOT NULL,  
    prenume VARCHAR2(50) NOT NULL,  
    id_agentie_turism INT,  
    nume_departament VARCHAR2(20) NOT NULL,  
    data_angajarii DATE,  
    FOREIGN KEY (id_agentie_turism) REFERENCES AGENTIE_TURISM (id_agentie_turism)  
);
```

```
CREATE TABLE CUMPARATOR (  
    id_cumparator INT PRIMARY KEY,  
    nume_cumparator VARCHAR2(20) NOT NULL,  
    prenume_cumparator VARCHAR2(50) NOT NULL,  
    numar_telefon VARCHAR2(11),  
    data_achizitie DATE  
);
```

```
CREATE TABLE BENEFICIU (  
    id_beneficiu INT PRIMARY KEY,  
    nume_beneficiu VARCHAR2(25) NOT NULL  
);
```

```
CREATE TABLE MOD_TRANSPORT (  
    id_mod_transport INT PRIMARY KEY,  
    denumire_transport VARCHAR2(20) NOT NULL  
);
```

```
CREATE TABLE TARA (  
    id_tara INT PRIMARY KEY,  
    nume_tara VARCHAR2(50) NOT NULL  
);
```

```
CREATE TABLE ORAS (  
    id_oras INT PRIMARY KEY,  
    nume_oras VARCHAR2(50) NOT NULL,  
    id_tara INT,  
    FOREIGN KEY (id_tara) REFERENCES TARA (id_tara)  
);
```

```
CREATE TABLE CAZARE (  
    id_cazare INT PRIMARY KEY,  
    nume_cazare VARCHAR2(50) NOT NULL,  
    id_oras INT,  
    FOREIGN KEY (id_oras) REFERENCES ORAS (id_oras)  
);
```

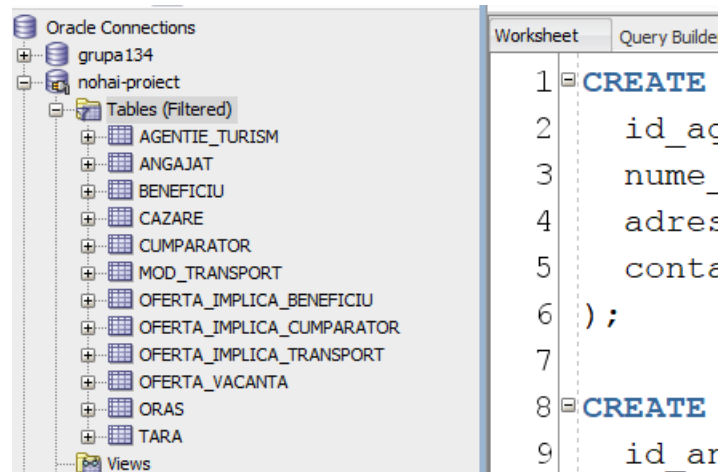
```
CREATE TABLE OFERTA_VACANTA (  
    id_oferta_vacanta INT PRIMARY KEY,  
    nume_oferta_vacanta VARCHAR2(50) NOT NULL,  
    id_tara INT,  
    id_agentie_turism INT,  
    FOREIGN KEY (id_tara) REFERENCES TARA (id_tara),  
    FOREIGN KEY (id_agentie_turism) REFERENCES AGENTIE_TURISM (id_agentie_turism)  
);
```

```
CREATE TABLE OFERTA_IMPLICA_BENEFICIU (  
    id_oferta_vacanta INT,  
    id_beneficiu INT,  
    PRIMARY KEY (id_oferta_vacanta, id_beneficiu),  
    FOREIGN KEY (id_oferta_vacanta) REFERENCES OFERTA_VACANTA  
(id_oferta_vacanta),  
    FOREIGN KEY (id_beneficiu) REFERENCES BENEFICIU (id_beneficiu)  
);
```

```
CREATE TABLE OFERTA_IMPLICA_TRANSPORT (  
    id_oferta_vacanta INT,  
    id_mod_transport INT,  
    pret INT,  
    PRIMARY KEY (id_oferta_vacanta, id_mod_transport),  
    FOREIGN KEY (id_oferta_vacanta) REFERENCES OFERTA_VACANTA  
(id_oferta_vacanta),  
    FOREIGN KEY (id_mod_transport) REFERENCES MOD_TRANSPORT (id_mod_transport)  
);
```

```
CREATE TABLE OFERTA_IMPLICA_CUMPARATOR (  
    id_oferta_vacanta INT,  
    id_cumparator INT,  
    PRIMARY KEY (id_oferta_vacanta, id_cumparator),  
    FOREIGN KEY (id_oferta_vacanta) REFERENCES OFERTA_VACANTA  
(id_oferta_vacanta),  
    FOREIGN KEY (id_cumparator) REFERENCES CUMPARATOR (id_cumparator)  
);
```





## Cerința 5

```
INSERT INTO TARA (id_tara, nume_tara)
VALUES (1, 'Romania');
```

```
INSERT INTO TARA (id_tara, nume_tara)
VALUES (2, 'Italia');
```

```
INSERT INTO TARA (id_tara, nume_tara)
VALUES (3, 'Grecia');
```

```
INSERT INTO TARA (id_tara, nume_tara)
VALUES (4, 'Thailanda');
```

```
INSERT INTO TARA (id_tara, nume_tara)
VALUES (5, 'Statele Unite');
```

ID_TARA	NUME_TARA
1	Romania
2	Italia
3	Grecia
4	Thailanda
5	Statele Unite

```
INSERT INTO ORAS (id_oras, nume_oras, id_tara)
VALUES (611, 'Bucuresti', 1);
```

```
INSERT INTO ORAS (id_oras, nume_oras, id_tara)
VALUES (612, 'Roma', 2);
```

```
INSERT INTO ORAS (id_oras, nume_oras, id_tara)
VALUES (613, 'Atena', 3);
```

```
INSERT INTO ORAS (id_oras, nume_oras, id_tara)
VALUES (614, 'Bangkok', 4);
```

```
INSERT INTO ORAS (id_oras, nume_oras, id_tara)
VALUES (615, 'New York', 5);
```

ID_ORAS	NUME_ORAS	ID_TARA
1	611 Bucuresti	1
2	612 Roma	2
3	613 Atena	3
4	614 Bangkok	4
5	615 New York	5

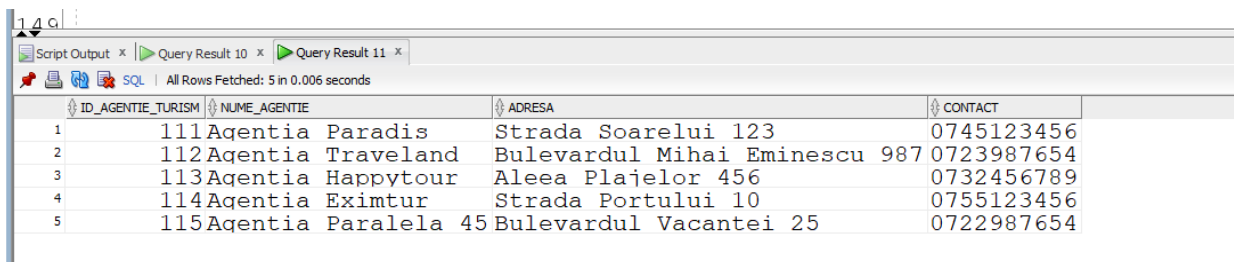
```
INSERT INTO AGENTIE_TURISM (id_agentie_turism, nume_agentie, adresa, contact)
VALUES (111, 'Agentia Paradis', 'Strada Soarelui 123', '0745123456');
```

```
INSERT INTO AGENTIE_TURISM (id_agentie_turism, nume_agentie, adresa, contact)
VALUES (112, 'Agentia Traveland', 'Bulevardul Mihai Eminescu 987', '0723987654');
```

```
INSERT INTO AGENTIE_TURISM (id_agentie_turism, nume_agentie, adresa, contact)
VALUES (113, 'Agentia Happytour', 'Aleea Plajelor 456', '0732456789');
```

```
INSERT INTO AGENTIE_TURISM (id_agentie_turism, nume_agentie, adresa, contact)
VALUES (114, 'Agentia Eximtur', 'Strada Portului 10', '0755123456');
```

```
INSERT INTO AGENTIE_TURISM (id_agentie_turism, nume_agentie, adresa, contact)
VALUES (115, 'Agentia Paralela 45', 'Bulevardul Vacantei 25', '0722987654');
```



The screenshot shows a database query result with 5 rows. The columns are ID\_AGENTIE\_TURISM, NUME\_AGENTIE, ADRESA, and CONTACT. The data is as follows:

ID_AGENTIE_TURISM	NUME_AGENTIE	ADRESA	CONTACT
1	111Agentia Paradis	Strada Soarelui 123	0745123456
2	112Agentia Traveland	Bulevardul Mihai Eminescu 987	0723987654
3	113Agentia Happytour	Aleea Plajelor 456	0732456789
4	114Agentia Eximtur	Strada Portului 10	0755123456
5	115Agentia Paralela 45	Bulevardul Vacantei 25	0722987654

```
INSERT INTO CAZARE (id_cazare, nume_cazare, id_oras)
VALUES (711, 'Hotel Paradise', 611);
```

```
INSERT INTO CAZARE (id_cazare, nume_cazare, id_oras)
VALUES (712, 'Villa Bella', 612);
```

```
INSERT INTO CAZARE (id_cazare, nume_cazare, id_oras)
VALUES (713, 'Beach Resort', 613);
```

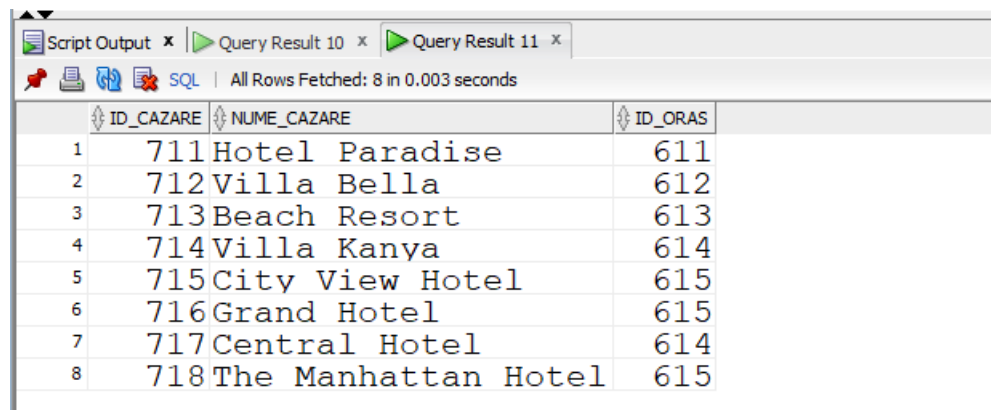
```
INSERT INTO CAZARE (id_cazare, nume_cazare, id_oras)
VALUES (714, 'Villa Kanya', 614);
```

```
INSERT INTO CAZARE (id_cazare, nume_cazare, id_oras)
VALUES (715, 'City View Hotel', 615);
```

```
INSERT INTO CAZARE (id_cazare, nume_cazare, id_oras)
VALUES (716, 'Grand Hotel', 615);
```

```
INSERT INTO CAZARE (id_cazare, nume_cazare, id_oras)
VALUES (717, 'Central Hotel', 614);
```

```
INSERT INTO CAZARE (id_cazare, nume_cazare, id_oras)
VALUES (718, 'The Manhattan Hotel', 615);
```



The screenshot shows a database query result window with the following data:

ID_CAZARE	NUME_CAZARE	ID_ORAS
711	Hotel Paradise	611
712	Villa Bella	612
713	Beach Resort	613
714	Villa Kanya	614
715	City View Hotel	615
716	Grand Hotel	615
717	Central Hotel	614
718	The Manhattan Hotel	615

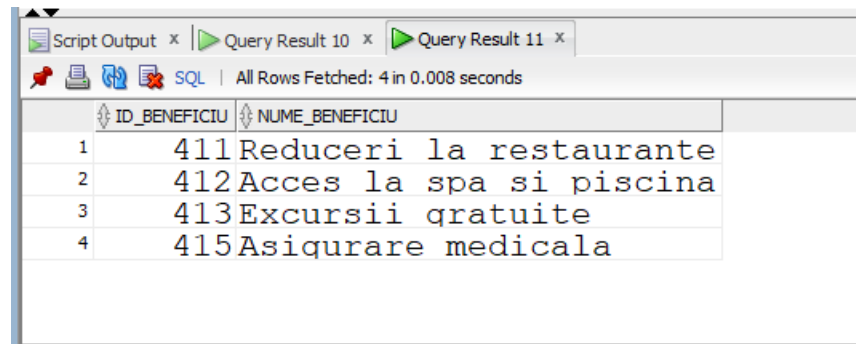
```
INSERT INTO BENEFICIU (id_beneficiu, nume_beneficiu)
VALUES (411, 'Reduceri la restaurante');
```

```
INSERT INTO BENEFICIU (id_beneficiu, nume_beneficiu)
VALUES (412, 'Acces la spa si piscina');
```

```
INSERT INTO BENEFICIU (id_beneficiu, nume_beneficiu)
VALUES (413, 'Excursii gratuite');
```

```
INSERT INTO BENEFICIU (id_beneficiu, nume_beneficiu)
VALUES (414, 'Serviciu de transfer inclus');
```

```
INSERT INTO BENEFICIU (id_beneficiu, nume_beneficiu)
VALUES (415, 'Asigurare medicala');
```



The screenshot shows a SQL query result in a database management tool. The window title is 'Script Output x | Query Result 10 x | Query Result 11 x'. The toolbar includes icons for saving, refreshing, and executing queries, along with a status bar indicating 'All Rows Fetched: 4 in 0.008 seconds'. The result is displayed in a table with two columns: 'ID\_BENEFICIU' and 'NUME\_BENEFICIU'. The data rows are as follows:

	ID_BENEFICIU	NUME_BENEFICIU
1	411	Reduceri la restaurante
2	412	Acces la spa si piscina
3	413	Excursii gratuite
4	415	Asiurare medicala

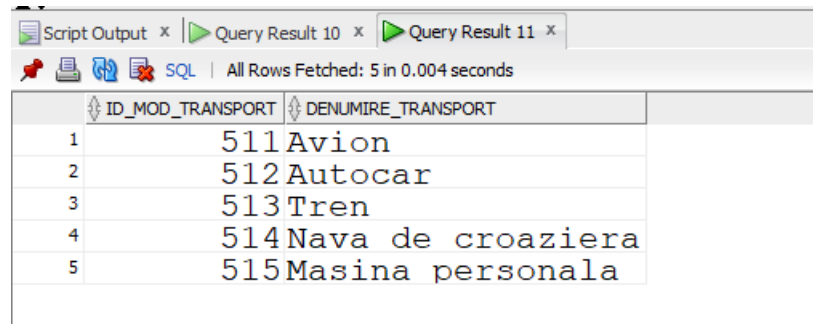
```
INSERT INTO MOD_TRANSPORT (id_mod_transport, denumire_transport)
VALUES (511, 'Avion');
```

```
INSERT INTO MOD_TRANSPORT (id_mod_transport, denumire_transport)
VALUES (512, 'Autocar');
```

```
INSERT INTO MOD_TRANSPORT (id_mod_transport, denumire_transport)
VALUES (513, 'Tren');
```

```
INSERT INTO MOD_TRANSPORT (id_mod_transport, denumire_transport)
VALUES (514, 'Nava de croaziera');
```

```
INSERT INTO MOD_TRANSPORT (id_mod_transport, denumire_transport)
VALUES (515, 'Masina personala');
```



The screenshot shows a database query result window with the following data:

ID_MOD_TRANSPORT	DENUMIRE_TRANSPORT
1	511 Avion
2	512 Autocar
3	513 Tren
4	514 Nava de croaziera
5	515 Masina personala

```
INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator,
numar_telefon , data_achizitie)
VALUES (seq_cumparator.nextval, 'Popa', 'Alexandra', '0745123456',TO_DATE('05-03-2022',
'DD-MM-YYYY'));
```

```
INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator,
numar_telefon, data_achizitie)
VALUES (seq_cumparator.nextval, 'Radu', 'Mihai', '0723987654',TO_DATE('03-07-2022', 'DD-
MM-YYYY'));
```

```
INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator,
numar_telefon , data_achizitie)
VALUES (seq_cumparator.nextval, 'Dumitrescu', 'George', '0732456789',TO_DATE('02-08-
2022', 'DD-MM-YYYY'));
```

```
INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator,
numar_telefon , data_achizitie)
VALUES (seq_cumparator.nextval, 'Georgescu', 'Andreea', '0754879632',TO_DATE('15-06-
2023', 'DD-MM-YYYY'));
```

```
INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator,
numar_telefon, data_achizitie)
```

```
VALUES (seq_cumparator.nextval, 'Stanciu', 'Ionut', '0765123789',TO_DATE('25-01-2023',
'DD-MM-YYYY'));
```

Script Output x Query Result x					
SQL   All Rows Fetched: 5 in 0 seconds					
	ID_CUMPARATOR	NUME_CUMPARATOR	PRENUME_CUMPARATOR	NUMAR_TELEFON	DATA_ACHIZITIE
1	811	Popa	Alexandra	0745123456	05-MAR-22
2	812	Radu	Mihai	0723987654	03-JUL-22
3	813	Dumitrescu	George	0732456789	02-AUG-22
4	814	Georgescu	Andreea	0754879632	15-JUN-23
5	815	Stanciu	Ionut	0765123789	25-JAN-23

```
INSERT INTO ANGAJAT (id_angajat, nume, prenume, id_agentie_turism, nume_departament,
data_angajarii)
```

```
VALUES (211, 'Popescu', 'Ion', 111, 'Vanzari', TO_DATE('2023-01-10', 'YYYY-MM-DD'));
```

```
INSERT INTO ANGAJAT (id_angajat, nume, prenume, id_agentie_turism, nume_departament,
data_angajarii)
```

```
VALUES (212, 'Ionescu', 'Maria', 111, 'Marketing', TO_DATE('2021-02-15', 'YYYY-MM-DD'));
```

```
INSERT INTO ANGAJAT (id_angajat, nume, prenume, id_agentie_turism, nume_departament,
data_angajarii)
```

```
VALUES (213, 'Mihai', 'Andrei', 112, 'Vanzari', TO_DATE('2022-03-20', 'YYYY-MM-DD'));
```

```
INSERT INTO ANGAJAT (id_angajat, nume, prenume, id_agentie_turism, nume_departament,
data_angajarii)
```

```
VALUES (214, 'Popa', 'Elena', 112, 'Administrativ', TO_DATE('2020-04-25', 'YYYY-MM-DD'));
```

```
INSERT INTO ANGAJAT (id_angajat, nume, prenume, id_agentie_turism, nume_departament,
data_angajarii)
```

```
VALUES (215, 'Stan', 'Ana', 113, 'Vanzari', TO_DATE('2022-05-30', 'YYYY-MM-DD'));
```

```
INSERT INTO ANGAJAT (id_angajat, nume, prenume, id_agentie_turism, nume_departament, data_angajarii)
```

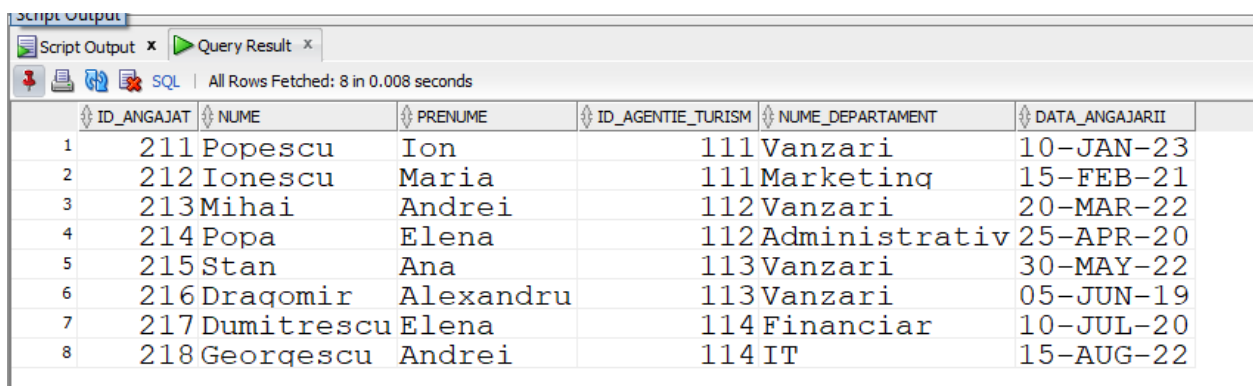
```
VALUES (216, 'Dragomir', 'Alexandru', 113, 'Vanzari', TO_DATE('2019-06-05', 'YYYY-MM-DD'));
```

```
INSERT INTO ANGAJAT (id_angajat, nume, prenume, id_agentie_turism, nume_departament, data_angajarii)
```

```
VALUES (217, 'Dumitrescu', 'Elena', 114, 'Financiar', TO_DATE('2020-07-10', 'YYYY-MM-DD'));
```

```
INSERT INTO ANGAJAT (id_angajat, nume, prenume, id_agentie_turism, nume_departament, data_angajarii)
```

```
VALUES (218, 'Georgescu', 'Andrei', 114, 'IT', TO_DATE('2022-08-15', 'YYYY-MM-DD'));
```



The screenshot shows a SQL query result window with the following data:

	ID_ANGAJAT	NUME	PRENUME	ID_AGENTIE_TURISM	NUME_DEPARTAMENT	DATA_ANGAJARII
1	211	Popescu	Ion	111	Vanzari	10-JAN-23
2	212	Ionescu	Maria	111	Marketing	15-FEB-21
3	213	Mihai	Andrei	112	Vanzari	20-MAR-22
4	214	Popa	Elena	112	Administrativ	25-APR-20
5	215	Stan	Ana	113	Vanzari	30-MAY-22
6	216	Dragomir	Alexandru	113	Vanzari	05-JUN-19
7	217	Dumitrescu	Elena	114	Financiar	10-JUL-20
8	218	Georgescu	Andrei	114	IT	15-AUG-22

```
INSERT INTO OFERTA_VACANTA (id_oferta_vacanta, nume_oferta_vacanta, id_tara, id_agentie_turism)
```

```
VALUES (311, 'Sejur exotic in Bali', 1, 111);
```

```
INSERT INTO OFERTA_VACANTA (id_oferta_vacanta, nume_oferta_vacanta, id_tara, id_agentie_turism)
```

```
VALUES (312, 'Circuit cultural prin Europa', 2, 112);
```



```
INSERT INTO OFERTA_VACANTA (id_oferta_vacanta, nume_oferta_vacanta, id_tara, id_agentie_turism)
```

```
VALUES (313, 'Vacanta la mare in Grecia', 3, 113);
```

```
INSERT INTO OFERTA_VACANTA (id_oferta_vacanta, nume_oferta_vacanta, id_tara, id_agentie_turism)
```

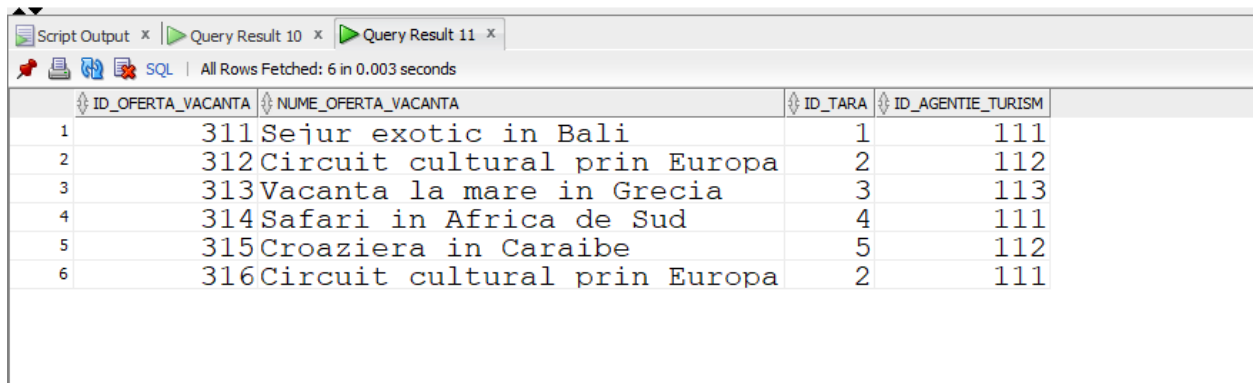
```
VALUES (314, 'Safari in Africa de Sud', 4, 111);
```

```
INSERT INTO OFERTA_VACANTA (id_oferta_vacanta, nume_oferta_vacanta, id_tara, id_agentie_turism)
```

```
VALUES (315, 'Croaziera in Caraibe', 5, 112);
```

```
INSERT INTO OFERTA_VACANTA (id_oferta_vacanta, nume_oferta_vacanta, id_tara, id_agentie_turism)
```

```
VALUES (316, 'Circuit cultural prin Europa', 2, 111);
```



The screenshot shows a database query result window with the following data:

ID_OFERTA_VACANTA	NUME_OFERTA_VACANTA	ID_TARA	ID_AGENTIE_TURISM
1	311Sejur exotic in Bali	1	111
2	312Circuit cultural prin Europa	2	112
3	313Vacanta la mare in Grecia	3	113
4	314Safari in Africa de Sud	4	111
5	315Croaziera in Caraibe	5	112
6	316Circuit cultural prin Europa	2	111

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
```

```
VALUES (311, 411);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
```

```
VALUES (311, 412);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
VALUES (312, 411);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
VALUES (312, 413);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
VALUES (313, 411);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
VALUES (313, 412);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
VALUES (313, 414);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
VALUES (314, 411);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
VALUES (314, 412);
```

```
INSERT INTO OFERTA_IMPLICA_BENEFICIU (id_oferta_vacanta, id_beneficiu)
VALUES (314, 414);
```

Query Result x		
All Rows Fetched: 10 in 0.012 seconds		
	ID_OFERTA_VACANTA	ID_BENEFICIU
1	311	411
2	311	412
3	312	411
4	312	413
5	313	411
6	313	412
7	313	414
8	314	411
9	314	412
10	314	414

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (311, 511, 500);
```

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (311, 512, 200);
```

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (312, 511, 800);
```

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (312, 513, 300);
```

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (313, 511, 600);
```

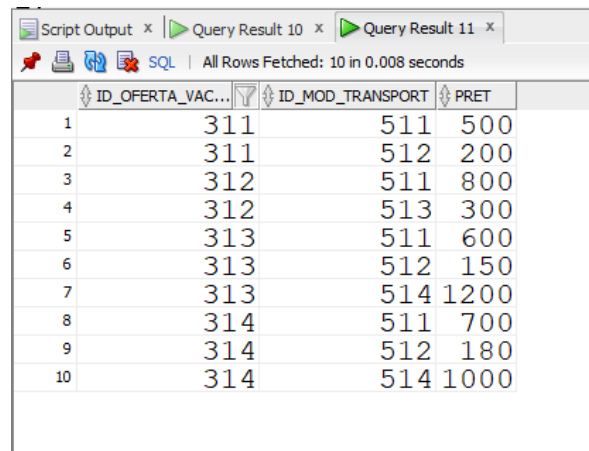
```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (313, 512, 150);
```

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (313, 514, 1200);
```

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (314, 511, 700);
```

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (314, 512, 180);
```

```
INSERT INTO OFERTA_IMPLICA_TRANSPORT (id_oferta_vacanta, id_mod_transport, pret)
VALUES (314, 514, 1000);
```



The screenshot shows a database query result window with the following data:

	ID_OFERTA_VAC...	ID_MOD_TRANSPORT	PRET
1	311	511	500
2	311	512	200
3	312	511	800
4	312	513	300
5	313	511	600
6	313	512	150
7	313	514	1200
8	314	511	700
9	314	512	180
10	314	514	1000

```
INSERT INTO OFERTA_IMPLICA_CUMPARATOR (id_oferta_vacanta, id_cumparator)
VALUES (311, 811);
```

```
INSERT INTO OFERTA_IMPLICA_CUMPARATOR (id_oferta_vacanta, id_cumparator)
VALUES (311, 812);
```

```
INSERT INTO OFERTA_IMPLICA_CUMPARATOR (id_oferta_vacanta, id_cumparator)
VALUES (312, 812);
```

```
INSERT INTO OFERTA_IMPLICA_CUMPARATOR (id_oferta_vacanta, id_cumparator)
```

VALUES (313, 813);

INSERT INTO OFERTA\_IMPLICA\_CUMPARATOR (id\_oferta\_vacanta, id\_cumparator)  
VALUES (313, 814);

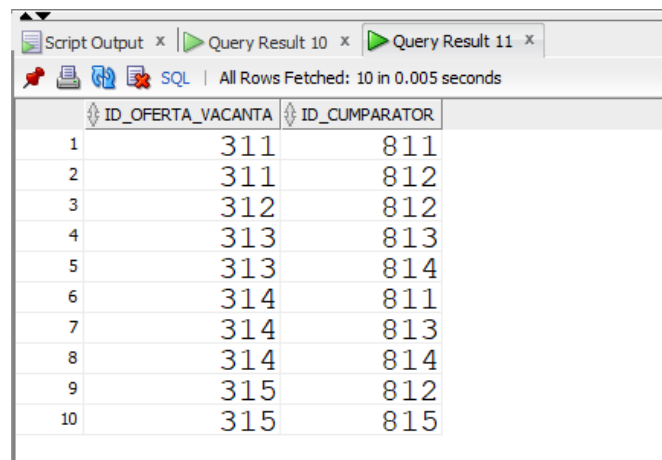
INSERT INTO OFERTA\_IMPLICA\_CUMPARATOR (id\_oferta\_vacanta, id\_cumparator)  
VALUES (314, 811);

INSERT INTO OFERTA\_IMPLICA\_CUMPARATOR (id\_oferta\_vacanta, id\_cumparator)  
VALUES (314, 813);

INSERT INTO OFERTA\_IMPLICA\_CUMPARATOR (id\_oferta\_vacanta, id\_cumparator)  
VALUES (314, 814);

INSERT INTO OFERTA\_IMPLICA\_CUMPARATOR (id\_oferta\_vacanta, id\_cumparator)  
VALUES (315, 812);

INSERT INTO OFERTA\_IMPLICA\_CUMPARATOR (id\_oferta\_vacanta, id\_cumparator)  
VALUES (315, 815);



The screenshot shows a database query result window with the following data:

	ID_OFERTA_VACANTA	ID_CUMPARATOR
1	311	811
2	311	812
3	312	812
4	313	813
5	313	814
6	314	811
7	314	813
8	314	814
9	315	812
10	315	815

## Cerința 6

Realizați o procedură stocată care primește ca parametru ID-ul unei oferte de vacanță si afișați următoarele informații pentru oferta respectivă:

- Numele ofertei de vacanță cu ID-ul furnizat.
- Lista beneficiilor implicate în oferta de vacanță.
- Lista beneficiilor care nu sunt implicate în oferta de vacanță.
- Lista clienților care au cumpărat oferta de vacanță, inclusiv informații precum ID-ul, numele, prenumele, numărul de telefon și data achiziției.

```
CREATE OR REPLACE PROCEDURE Cerinta6(
    in_id_oferta IN OFERTA_VACANTA.id_oferta_vacanta%TYPE
)
IS
    -- Record pentru a memora informatiile despre beneficii
    TYPE infobeneficii IS RECORD
    (
        beneficiu_id OFERTA_IMPLICA_BENEFICIU.id_beneficiu%TYPE,
        nume        BENEFICIU.nume_beneficiu%TYPE
    );

    TYPE client_info IS RECORD (
        id_cumparator CUMPARATOR.id_cumparator%TYPE,
        nume_cumparator CUMPARATOR.nume_cumparator%TYPE,
        prenume_cumparator CUMPARATOR.prenume_cumparator%TYPE,
        numar_telefon CUMPARATOR.numar_telefon%TYPE,
        data_achizitie CUMPARATOR.data_achizitie%TYPE
    );

    nume_oferta    VARCHAR2(100); -- Numele ofertei cu ID-ul in_id_oferta
```

```

-- Tabel indexat pentru memorarea beneficiilor implicate in oferta
TYPE beneficii_index_table IS TABLE OF infobeneficii INDEX BY PLS_INTEGER;
t_beneficii_index beneficii_index_table;

-- Tabel imbricat pentru a memora beneficiile care NU sunt implicate in oferta
TYPE beneficii_imbricat IS TABLE OF infobeneficii;
t_beneficii_imbricat beneficii_imbricat := beneficii_imbricat();

-- VARRAY pentru a memora informatiile despre clienti
TYPE clienti_varray IS VARRAY(100) OF client_info;
t_clienti clienti_varray := clienti_varray();
BEGIN
    BEGIN
        -- Verificare daca exista oferta de vacanta cu ID-ul dat
        SELECT nume_oferta_vacanta
        INTO nume_oferta
        FROM oferta_vacanta
        WHERE id_oferta_vacanta = in_id_oferta;

        EXCEPTION
        WHEN no_data_found THEN
            dbms_output.put_line('Oferta de vacanta cu ID-ul ' || in_id_oferta || ' nu exista.');
```

RETURN;

```

    END;

---memorare informatii despre beneficiile implicate in oferta de vacanta in tabelul indexat
BEGIN
    SELECT ob.id_beneficiu, b.nume_beneficiu BULK COLLECT
```

```
INTO t_beneficii_index
```

```
FROM oferta_implica_beneficiu ob
```

```
    JOIN beneficiu b ON ob.id_beneficiu = b.id_beneficiu
```

```
WHERE ob.id_oferta_vacanta = in_id_oferta;
```

```
---memorare in tabelul imbricat a beneficiilor care nu sunt implicate in oferta de vacanta
```

```
FOR i IN (SELECT b.id_beneficiu, b.nume_beneficiu
```

```
    FROM beneficiu b
```

```
    WHERE b.id_beneficiu NOT IN (SELECT ob1.id_beneficiu
```

```
        FROM oferta_implica_beneficiu ob1
```

```
        WHERE ob1.id_oferta_vacanta = in_id_oferta))
```

```
LOOP
```

```
    t_beneficii_imbricat.extend;
```

```
    t_beneficii_imbricat(t_beneficii_imbricat.LAST).beneficiu_id := i.id_beneficiu;
```

```
    t_beneficii_imbricat(t_beneficii_imbricat.LAST).nume := i.nume_beneficiu;
```

```
END LOOP;
```

```
dbms_output.put_line('-----');
```

```
dbms_output.put_line(nume_oferta || ':');
```

```
dbms_output.put_line('-----');
```

```
IF t_beneficii_index.COUNT = 0 THEN
```

```
    dbms_output.put_line('Nu exista beneficii pentru oferta de vacanta cu ID-ul ' ||  
in_id_oferta);
```

```
    dbms_output.put_line('-----');
```

```
ELSE
```

```
    dbms_output.put_line('Beneficii implicate:');
```

```
    FOR i IN t_beneficii_index.FIRST..t_beneficii_index.LAST
```



```

LOOP
    dbms_output.put_line(' ' || t_beneficii_index(i).nume);
END LOOP;
END IF;

IF t_beneficii_imbricat.COUNT = 0 THEN
    dbms_output.put_line('TOATE BENEFICIILE SUNT IMPLICATE');
    dbms_output.put_line('-----');
ELSE
    dbms_output.put_line('-----');
    dbms_output.put_line('Beneficii care nu sunt implicate:');

    FOR i IN t_beneficii_imbricat.FIRST..t_beneficii_imbricat.LAST
    LOOP
        dbms_output.put_line(' ' || t_beneficii_imbricat(i).nume);
    END LOOP;

```

```

END IF;

```

```

--Memorarea clientilor care au achizitionat oferta de vacanta

```

```

SELECT c.id_cumparator, c.nume_cumparator, c.preume_cumparator, c.numar_telefon,
c.data_achizitie

```

```

BULK COLLECT INTO t_clienti

```

```

FROM oferta_implica_cumparator oc

```

```

    JOIN cumparator c ON oc.id_cumparator = c.id_cumparator

```

```

WHERE oc.id_oferta_vacanta = in_id_oferta;

```

```

dbms_output.put_line('-----');

```

```

IF t_clienti.COUNT = 0 THEN

```

```

    dbms_output.put_line('Nu exista clienti care au achizitionat oferta.');
```

```

ELSE
    dbms_output.put_line('Informatii despre clientii care au cumparat oferta:');
    FOR i IN 1..t_clienti.COUNT
    LOOP
        dbms_output.put_line('ID: ' || t_clienti(i).id_cumparator ||
                               ', Nume: ' || t_clienti(i).nume_cumparator ||
                               ' ' || t_clienti(i).prenume_cumparator ||
                               ', Telefon: ' || t_clienti(i).numar_telefon ||
                               ', Data Achizitie: ' || t_clienti(i).data_achizitie);
    END LOOP;
END IF;
dbms_output.put_line('-----');
END;
END Cerinta6;
/

-- Apelul subprogramului
BEGIN
    Cerinta6('314');
END;
/

```

```
Gr2341.sql Welcome Page Gr234 nohai-project
Worksheet Query Builder
119 dbms_output.put_line('-----');
120 END;
121 END Cerinta6;
122 /
123
124 -- Apelul subprogramului
125 BEGIN
126 Cerinta6('314');
127 END;
Script Output x
Task completed in 0.089 seconds

-----
Safari in Africa de Sud:
-----
Beneficii implicate:
Reduceri la restaurante
Acces la spa si piscina
Serviciu de transfer inclus
-----
Beneficii care nu sunt implicate:
Excursii gratuite
Asigurare medicala
-----
Informatii despre clientii care au cumparat oferta:
ID: 811, Nume: Popa Alexandra, Telefon: 0745123456, Data Achizitie: 05-MAR-22
ID: 813, Nume: Dumitrescu George, Telefon: 0732456789, Data Achizitie: 02-AUG-22
ID: 814, Nume: Georgescu Andreea, Telefon: 0754879632, Data Achizitie: 15-JUN-23
-----
```

## Cerința 7

Pentru un id-ul unei oferte de vacanta trimis ca parametru, afisati detalii despre agentii de turism a fiecarei oferte care are id-ul mai mic decat cel trimis ca parametru.

```
CREATE OR REPLACE PROCEDURE Cerinta7(
    in_oferta_id IN OFERTA_VACANTA.id_oferta_vacanta%TYPE
)
IS
    -- Declaram un cursor implicit pentru informatiile despre oferta de vacanta
    CURSOR oferta_cursor IS
        SELECT id_oferta_vacanta, nume_oferta_vacanta, id_agentie_turism
        FROM oferta_vacanta
        WHERE id_oferta_vacanta < in_oferta_id;

    -- Declaram un cursor parametrizat dependent de primul cursor
```

```

CURSOR agentie_cursor(p_cursor_oferta IN oferta_cursor%ROWTYPE) IS

    SELECT id_agentie_turism, nume_agentie, adresa

    FROM agentie_turism

    WHERE id_agentie_turism = p_cursor_oferta.id_agentie_turism;

v_oferta_rec oferta_cursor%ROWTYPE;
v_agentie_rec agentie_cursor%ROWTYPE;

BEGIN

    OPEN oferta_cursor;

    LOOP

        FETCH oferta_cursor INTO v_oferta_rec;

        EXIT WHEN oferta_cursor%NOTFOUND;

        DBMS_OUTPUT.PUT_LINE('Oferta ID: ' || v_oferta_rec.id_oferta_vacanta || ', Nume: ' ||
v_oferta_rec.nume_oferta_vacanta);

        -- Deschidem al doilea cursor, transmitem rezultatele primului ca parametru
        OPEN agentie_cursor(v_oferta_rec);

        FETCH agentie_cursor INTO v_agentie_rec;

        IF agentie_cursor%FOUND THEN

            DBMS_OUTPUT.PUT_LINE(' Agentie ID: ' || v_agentie_rec.id_agentie_turism || ',
Nume: ' || v_agentie_rec.nume_agentie || ', Adresa: ' || v_agentie_rec.adresa);

            END IF;

        CLOSE agentie_cursor;

```

```
DBMS_OUTPUT.PUT_LINE('-----');

END LOOP;

CLOSE oferta_cursor;

END Cerinta7;

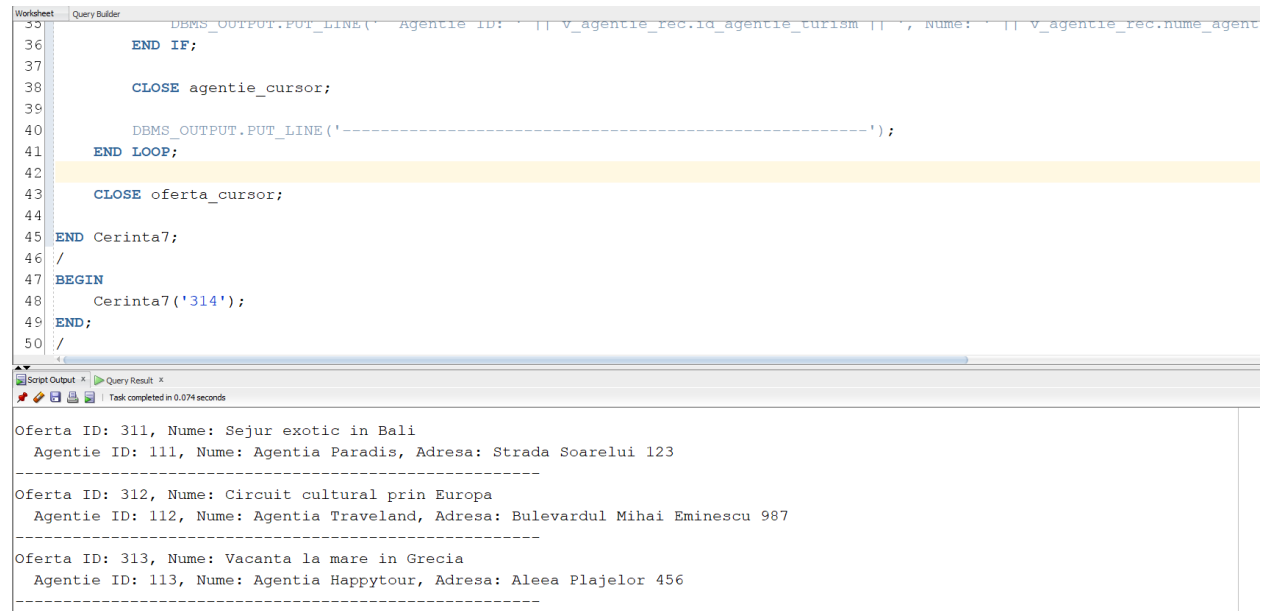
/

BEGIN

    Cerinta7('314');

END;

/
```



```
DBMS_OUTPUT.PUT_LINE('Agentie ID: ' || v_agentie_rec.id_agentie_turism || ', Nume: ' || v_agentie_rec.num_agentie_turism || ', Adresa: ' || v_agentie_rec.adresa);
```

```
36      END IF;
```

```
37
```

```
38      CLOSE agentie_cursor;
```

```
39
```

```
40      DBMS_OUTPUT.PUT_LINE('-----');
```

```
41  END LOOP;
```

```
42
```

```
43  CLOSE oferta_cursor;
```

```
44
```

```
45  END Cerinta7;
```

```
46  /
```

```
47  BEGIN
```

```
48      Cerinta7('314');
```

```
49  END;
```

```
50  /
```

Script Output x Query Result x

Task completed in 0.074 seconds

Oferta ID: 311, Nume: Sejur exotic in Bali  
Agentie ID: 111, Nume: Agentia Paradis, Adresa: Strada Soarelui 123  
-----

Oferta ID: 312, Nume: Circuit cultural prin Europa  
Agentie ID: 112, Nume: Agentia Traveland, Adresa: Bulevardul Mihai Eminescu 987  
-----

Oferta ID: 313, Nume: Vacanta la mare in Grecia  
Agentie ID: 113, Nume: Agentia Happytour, Adresa: Aleea Plajelor 456  
-----

## Cerința 8

Creați o procedură stocată care, primind un ID de cumpărător, să afișeze:

- Numărul total de vacanțe cu beneficii achiziționate de către acel cumpărător.

Procedura ar trebui să gestioneze cazurile în care:

- Parametrul nu este un număr întreg pozitiv.

- Nu există date pentru cumpărătorul respectiv.

```
CREATE OR REPLACE FUNCTION Cerinta8(in_id_cumparator IN  
CUMPARATOR.id_cumparator%TYPE) RETURN NUMBER
```

```
IS
```

```
    v_numar_vacante NUMBER;
```

```
    NoDataFoundException EXCEPTION;
```

```
    InvalidInputException EXCEPTION;
```

```
BEGIN
```

```
    -- Verificam daca parametrul este un numar intreg pozitiv
```

```
    IF in_id_cumparator IS NULL OR in_id_cumparator <= 0 THEN
```

```
        RAISE InvalidInputException;
```

```
    END IF;
```

```
    v_numar_vacante := 0;
```

```
    SELECT COUNT(DISTINCT oic.id_oferta_vacanta)
```

```
    INTO v_numar_vacante
```

```
    FROM OFERTA_IMPLICA_CUMPARATOR oic
```

```
    JOIN OFERTA_IMPLICA_BENEFICIU oib ON oic.id_oferta_vacanta =  
oib.id_oferta_vacanta
```

```
    WHERE oic.id_cumparator = in_id_cumparator;
```

```
    IF v_numar_vacante > 0 THEN
```

```
        RETURN v_numar_vacante;
```

```
    ELSE
```

```
        RAISE NoDataFoundException;
```

```
    END IF;
```

EXCEPTION

WHEN NoDataFoundException THEN

DBMS\_OUTPUT.PUT\_LINE('Nu s-au gasit date pentru cumparatorul cu ID-ul: ' ||  
in\_id\_cumparator);

RETURN -1;

WHEN InvalidInputException THEN

DBMS\_OUTPUT.PUT\_LINE('Parametrul trebuie sa fie un numar intreg pozitiv.');

RETURN -1;

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('A aparut o eroare neasteptata.');

RETURN -1;

END Cerinta8;

/

DECLARE

exemplu NUMBER;

BEGIN

exemplu := cerinta8(811);

IF exemplu > -1 THEN

DBMS\_OUTPUT.PUT\_LINE('Numarul total de vacante cu beneficii achizitionate: ' ||  
exemplu);

END IF;

END;

/

```

36 END Cerinta8;
37 /
38
39 DECLARE
40     exemplu NUMBER;
41 BEGIN
42     exemplu := cerinta8(811);
43     IF exemplu > -1 THEN
44         DBMS_OUTPUT.PUT_LINE('Numarul total de vacante cu beneficii achizitionate: ' || exemplu);
45     END IF;
46 END;
47 /
48
49

```

Script Output x Task completed in 0.082 seconds

PL/SQL procedure successfully completed.

Numarul total de vacante cu beneficii achizitionate: 2

```

39 DECLARE
40     exemplu NUMBER;
41 BEGIN
42     exemplu := cerinta8(10);
43     IF exemplu > -1 THEN
44         DBMS_OUTPUT.PUT_LINE('Numarul total de vacante cu beneficii achizitionate: ' || exemplu);
45     END IF;
46 END;
47 /
48
49

```

Script Output x Task completed in 0.086 seconds

PL/SQL procedure successfully completed.

Nu s-au gasit date pentru cumparatorul cu ID-ul: 10

PL/SQL procedure successfully completed.

```

38
39 DECLARE
40     exemplu NUMBER;
41 BEGIN
42     exemplu := cerinta8(-19);
43     IF exemplu > -1 THEN
44         DBMS_OUTPUT.PUT_LINE('Numarul total de vacante cu beneficii achizitionate: ' || exemplu);
45     END IF;
46 END;
47 /
48
49

```

Script Output x Task completed in 0.073 seconds

PL/SQL procedure successfully completed.

Parametrul trebuie sa fie un numar intreg pozitiv.

PL/SQL procedure successfully completed.



## Cerința 9

Afisati informatii legate de modalitatea de transport care ii este oferita clientului, in momentul in care vrea sa cumpere o oferta de vacanta dintr-o anumita tara. Tratati exceptiile astfel incat intr-o anumita tara sa nu existe mai mult de 3 moduri de transport.

```
CREATE OR REPLACE PROCEDURE Cerinta9(p_nume_tara VARCHAR2) AS
```

```
    tara_not_found EXCEPTION;
```

```
BEGIN
```

```
    DECLARE
```

```
        v_count NUMBER;
```

```
BEGIN
```

```
-- Verifica daca exista tara în tabelul de tari
```

```
SELECT COUNT(*)
```

```
    INTO v_count
```

```
    FROM TARA t
```

```
    WHERE UPPER(t.nume_tara) = UPPER(p_nume_tara);
```

```
IF v_count = 0 THEN
```

```
    RAISE tara_not_found;
```

```
END IF;
```

```
-- Verifică dacă există date pentru țara specificată
```

```
SELECT COUNT(DISTINCT mt.denumire_transport)
```

```
    INTO v_count
```

```
    FROM TARA t
```

```
    JOIN ORAS o ON t.id_tara = o.id_tara
```

```
    JOIN CAZARE c ON o.id_oras = c.id_oras
```

```
    JOIN OFERTA_VACANTA ov ON t.id_tara = ov.id_tara
```

```

    JOIN OFERTA_IMPLICA_TRANSPORT oit ON ov.id_oferta_vacanta =
oit.id_oferta_vacanta

    JOIN MOD_TRANSPORT mt ON oit.id_mod_transport = mt.id_mod_transport
    WHERE UPPER(t.ume_tara) = UPPER(p_ume_tara);

IF v_count = 0 THEN
    RAISE NO_DATA_FOUND;
ELSIF v_count > 3 THEN
    RAISE TOO_MANY_ROWS;
END IF;

-- Afisează rezultatele
FOR modalitate IN (SELECT c.ume_cazare,
                        o.ume_oras,
                        t.ume_tara,
                        LISTAGG(mt.denumire_transport, ' sau ') WITHIN GROUP (ORDER BY
mt.denumire_transport) AS modalitati_transport
FROM TARA t
JOIN ORAS o ON t.id_tara = o.id_tara
JOIN CAZARE c ON o.id_oras = c.id_oras
JOIN OFERTA_VACANTA ov ON t.id_tara = ov.id_tara
JOIN OFERTA_IMPLICA_TRANSPORT oit ON ov.id_oferta_vacanta =
oit.id_oferta_vacanta
JOIN MOD_TRANSPORT mt ON oit.id_mod_transport = mt.id_mod_transport
WHERE UPPER(t.ume_tara) = UPPER(p_ume_tara)
GROUP BY c.ume_cazare, o.ume_oras, t.ume_tara)
LOOP
    DBMS_OUTPUT.PUT_LINE('Pentru a ajunge la ' || modalitate.ume_cazare || ', ' ||
        modalitate.ume_oras || ', ' || modalitate.ume_tara ||

```

```

        ', vei folosi mijloacele de transport: ' || modalitate.modalitati_transport);
    END LOOP;
EXCEPTION
    WHEN tara_not_found THEN
        DBMS_OUTPUT.PUT_LINE('Nu exista oferte de vacanta cu destinatia: ' || p_nume_tara);
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Nu exista modalitati de transport incluse in ofertele de
vacanta din tara: ' || p_nume_tara);
    WHEN TOO_MANY_ROWS THEN
        DBMS_OUTPUT.PUT_LINE('Prea multe date pentru tara specificata. ');
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('A aparut o eroare neasteptata. ');
    END;
END Cerinta9;

/

BEGIN
    Cerinta9('Thailanda');
    Cerinta9('Statele Unite');
    Cerinta9('Grecia');
    Cerinta9('Japonia');
END;

/

select * from tara;

```

```
66 BEGIN
```

```
67   Cerinta9('Thailanda');
```

```
68   --Cerinta9('Statele Unite');
```

```
69   --Cerinta9('Grecia');
```

```
70 END;
```

PL/SQL procedure successfully completed.

Pentru a ajunge la Villa Kanya, Bangkok, Thailanda, vei folosi mijloacele de transport: Autocar sau Avion sau Nava de croaziera  
Pentru a ajunge la Central Hotel, Bangkok, Thailanda, vei folosi mijloacele de transport: Autocar sau Avion sau Nava de croaziera

```
66 BEGIN
```

```
67   --Cerinta9('Thailanda');
```

```
68   Cerinta9('Statele Unite');
```

```
69   --Cerinta9('Grecia');
```

```
70 END;
```

PL/SQL procedure successfully completed.

Nu exista modalitati de transport incluse in ofertele de vacanta din tara: Statele Unite

PL/SQL procedure successfully completed.

```
66 BEGIN
```

```
67   --Cerinta9('Thailanda');
```

```
68   --Cerinta9('Statele Unite');
```

```
69   Cerinta9('Grecia');
```

```
70 END;
```

Procedure CERINTIA9 compiled

Prea multe date pentru tara specificata.

PL/SQL procedure successfully completed.

```
66 BEGIN
```

```
67   --Cerinta9('Thailanda');
```

```
68   --Cerinta9('Statele Unite');
```

```
69   --Cerinta9('Grecia');
```

```
70   Cerinta9('Japonia');
```

```
71 END;
```

Prea multe date pentru tara specificata.

PL/SQL procedure successfully completed.

Nu exista oferte de vacanta cu destinatia: Japonia

## Cerința 10

Trigger pentru a nu permite inserarea a mai mult de 5 agentii in sistemul de gestiune.

```
CREATE OR REPLACE TRIGGER Cerinta10
```

```
BEFORE INSERT ON AGENTIE_TURISM
```

```
DECLARE
```

```
    v_nr_agentii NUMBER;
```

```
BEGIN
```

```
    SELECT COUNT(id_agentie_turism)
```

```
    INTO v_nr_agentii
```

```
    FROM AGENTIE_TURISM;
```

```
    IF v_nr_agentii >= 5 THEN
```

```
        -- Afișați un mesaj specific și aruncați o excepție
```

```
        RAISE_APPLICATION_ERROR(-20001, 'Fonduri Insuficiente pentru gestionarea inca  
unei Agentii de Turism!');
```

```
    END IF;
```

```
END;
```

```
/
```

```
INSERT INTO AGENTIE_TURISM (id_agentie_turism, nume_agentie, adresa, contact)
```

```
VALUES (117, 'Agentia Eximturs', 'Strada Portului 10', '0755123456');
```

```
16 INSERT INTO AGENTIE_TURISM (id_agentie_turism, nume_agentie, adresa, contact)  
17 VALUES (117, 'Agentia Eximturs', 'Strada Portului 10', '0755123456');  
18
```

```
Script Output: X | Query Result: X | Query Result 1: X  
Task completed in 0.071 seconds  
INSERT INTO AGENTIE_TURISM (id_agentie_turism, nume_agentie, adresa, contact)  
VALUES (117, 'Agentia Eximturs', 'Strada Portului 10', '0755123456')  
Error report -  
ORA-20001: Fonduri Insuficiente pentru gestionarea inca unei Agentii de Turism!  
ORA-06512: at "NOHAI.RESTRICTIE_ANGAJATI_AGENTIE", line 10  
ORA-04088: error during execution of trigger 'NOHAI.RESTRICTIE_ANGAJATI_AGENTIE'
```

## Cerința 11

Trigger pentru a nu permite inserarea unui cumparator care cumpara o oferta in zilele de luni, in prima zi a lunii, sau in anul 2020.

```
CREATE OR REPLACE TRIGGER Cerinta11
BEFORE INSERT OR UPDATE ON CUMPARATOR
FOR EACH ROW
DECLARE
    ziua_saptamanii VARCHAR2(10);
    zi_luna NUMBER;
    anul NUMBER;
BEGIN

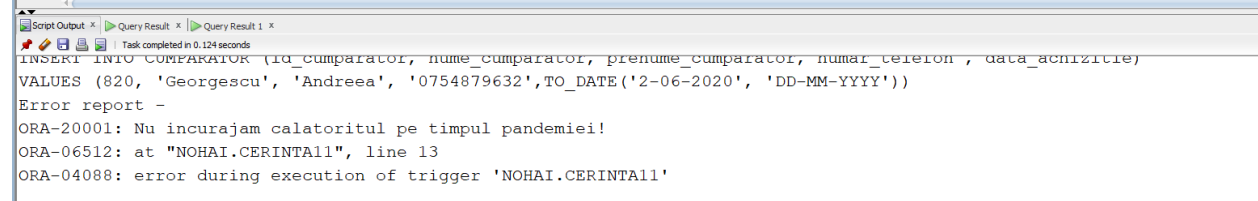
    ziua_saptamanii := TO_CHAR(:NEW.data_achizitie, 'DAY');
    zi_luna := TO_NUMBER(TO_CHAR(:NEW.data_achizitie, 'DD'));
    anul := TO_NUMBER(TO_CHAR(:NEW.data_achizitie, 'YYYY'));

    IF ziua_saptamanii = 'MONDAY' OR zi_luna = 1 OR anul = 2020 THEN
        IF anul = 2020 THEN
            RAISE_APPLICATION_ERROR(-20001, 'Nu incurajam calatoritul pe timpul
pandemiei!');
        ELSE
            RAISE_APPLICATION_ERROR(-20001, 'Recomandam sa nu cheltuiti banii in zilele de
luni sau de inceput de luna! Ne veti multumi!');
        END IF;
    END IF;
END;
/
```

INSERT INTO CUMPARATOR (id\_cumparator, nume\_cumparator, prenume\_cumparator, numar\_telefon , data\_achizitie)

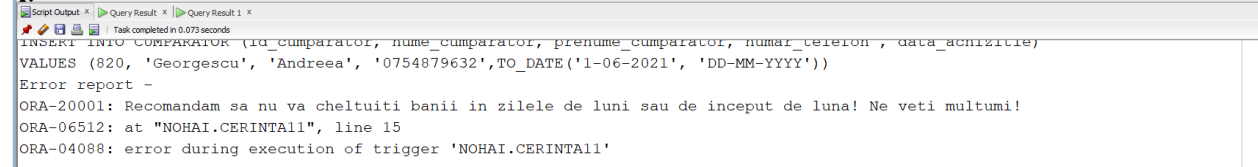
VALUES (820, 'Georgescu', 'Andreea', '0754879632',TO\_DATE('2-06-2020', 'DD-MM-YYYY'));

```
25
26 INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator, numar_telefon , data_achizitie)
27 VALUES (820, 'Georgescu', 'Andreea', '0754879632',TO_DATE('2-06-2020', 'DD-MM-YYYY'));
28
29
```



```
Script Output x Query Result x Query Result 1 x
Task completed in 0.124 seconds
INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator, numar_telefon , data_achizitie)
VALUES (820, 'Georgescu', 'Andreea', '0754879632',TO_DATE('2-06-2020', 'DD-MM-YYYY'))
Error report -
ORA-20001: Nu incurajam calatoritul pe timpul pandemiei!
ORA-06512: at "NOHAI.CERINTAI1", line 13
ORA-04088: error during execution of trigger 'NOHAI.CERINTAI1'
```

```
26 INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator, numar_telefon , data_achizitie)
27 VALUES (820, 'Georgescu', 'Andreea', '0754879632',TO_DATE('1-06-2021', 'DD-MM-YYYY'));
28
29
```



```
Script Output x Query Result x Query Result 1 x
Task completed in 0.073 seconds
INSERT INTO CUMPARATOR (id_cumparator, nume_cumparator, prenume_cumparator, numar_telefon , data_achizitie)
VALUES (820, 'Georgescu', 'Andreea', '0754879632',TO_DATE('1-06-2021', 'DD-MM-YYYY'))
Error report -
ORA-20001: Recomandam sa nu va cheltuiti banii in zilele de luni sau de inceput de luna! Ne veti multumi!
ORA-06512: at "NOHAI.CERINTAI1", line 15
ORA-04088: error during execution of trigger 'NOHAI.CERINTAI1'
```

## Cerinta 12

Trigger care se declanseaza la fiecare operatie de tip LDD si care insereaza in tabela toate modificarile facute.

```
CREATE TABLE informatii (
baza_date VARCHAR2(40),
nume_utilizator VARCHAR2(40),
modificare VARCHAR2(40),
data_modificare DATE);
```

```
CREATE OR REPLACE TRIGGER cerinta12
```

```
AFTER CREATE OR DROP OR ALTER ON SCHEMA
```

BEGIN

INSERT INTO informatii VALUES(SYS.database\_name, SYS.login\_user, SYS.sysevent, sysdate);

END;

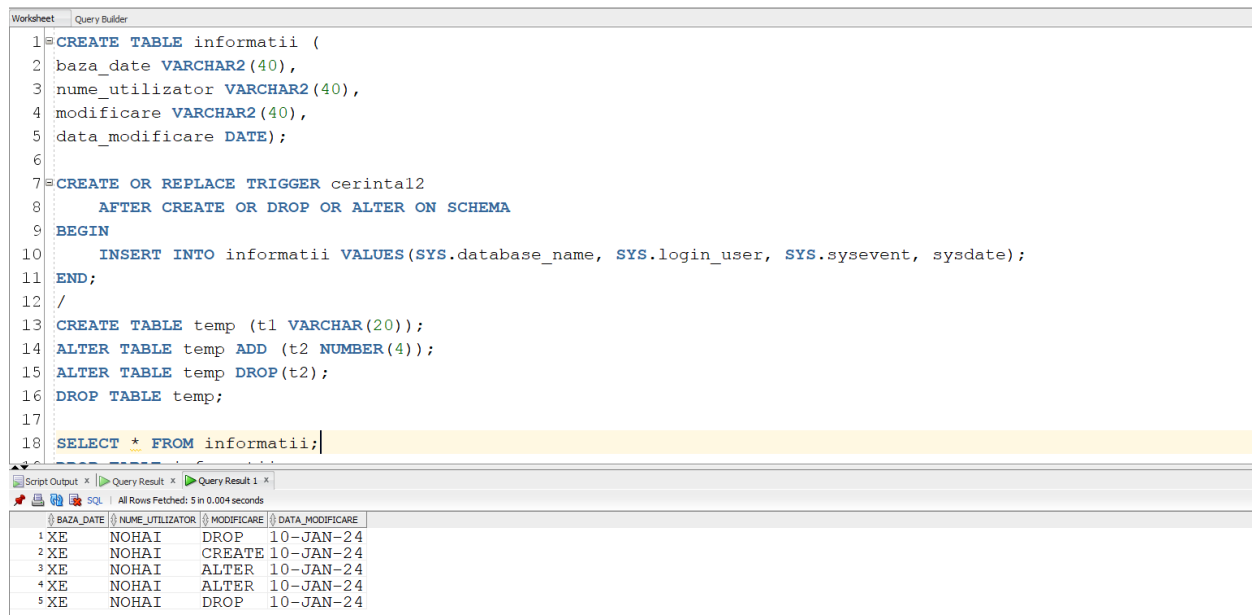
/

CREATE TABLE temp (t1 VARCHAR(20));

ALTER TABLE temp ADD (t2 NUMBER(4));

ALTER TABLE temp DROP(t2);

DROP TABLE temp;



The screenshot shows a SQL Developer window with a script editor and a query result table. The script editor contains the following SQL code:

```
1 CREATE TABLE informatii (  
2 baza_date VARCHAR2(40),  
3 nume_utilizator VARCHAR2(40),  
4 modificare VARCHAR2(40),  
5 data_modificare DATE);  
6  
7 CREATE OR REPLACE TRIGGER cerinta12  
8 AFTER CREATE OR DROP OR ALTER ON SCHEMA  
9 BEGIN  
10 INSERT INTO informatii VALUES(SYS.database_name, SYS.login_user, SYS.sysevent, sysdate);  
11 END;  
12 /  
13 CREATE TABLE temp (t1 VARCHAR(20));  
14 ALTER TABLE temp ADD (t2 NUMBER(4));  
15 ALTER TABLE temp DROP(t2);  
16 DROP TABLE temp;  
17  
18 SELECT * FROM informatii;
```

The query result table shows the following data:

	BAZA_DATE	NUME_UTILIZATOR	MODIFICARE	DATA_MODIFICARE
1	XE	NOHAI	DROP	10-JAN-24
2	XE	NOHAI	CREATE	10-JAN-24
3	XE	NOHAI	ALTER	10-JAN-24
4	XE	NOHAI	ALTER	10-JAN-24
5	XE	NOHAI	DROP	10-JAN-24

## Cerința 13

Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului.

CREATE OR REPLACE PACKAGE cerinta13 AS

PROCEDURE Cerinta6(in\_id\_oferta IN OFERTA\_VACANTA.id\_oferta\_vacanta%TYPE);

PROCEDURE Cerinta7(in\_oferta\_id IN OFERTA\_VACANTA.id\_oferta\_vacanta%TYPE);

FUNCTION Cerinta8(in\_id\_cumparator IN CUMPARATOR.id\_cumparator%TYPE)  
RETURN NUMBER;

PROCEDURE Cerinta9(p\_nume\_tara VARCHAR2);

END cerinta13;



/

CREATE OR REPLACE PACKAGE BODY cerinta13 AS

PROCEDURE Cerinta6(in\_id\_oferta IN OFERTA\_VACANTA.id\_oferta\_vacanta%TYPE)

IS

-- Record pentru a memora informatiile despre beneficii

TYPE infobeneficii IS RECORD

(

beneficiu\_id OFERTA\_IMPLICA\_BENEFICIU.id\_beneficiu%TYPE,

nume BENEFICIU.nume\_beneficiu%TYPE

);

TYPE client\_info IS RECORD (

id\_cumparator CUMPARATOR.id\_cumparator%TYPE,

nume\_cumparator CUMPARATOR.nume\_cumparator%TYPE,

prenume\_cumparator CUMPARATOR.prenume\_cumparator%TYPE,

numar\_telefon CUMPARATOR.numar\_telefon%TYPE,

data\_achizitie CUMPARATOR.data\_achizitie%TYPE

);

nume\_oferta VARCHAR2(100); -- Numele ofertei cu ID-ul in\_id\_oferta

-- Tabel indexat pentru memorarea beneficiilor implicate in oferta

TYPE beneficii\_index\_table IS TABLE OF infobeneficii INDEX BY PLS\_INTEGER;

t\_beneficii\_index beneficii\_index\_table;

-- Tabel imbricat pentru a memora beneficiile care NU sunt implicate in oferta

TYPE beneficii\_imbricat IS TABLE OF infobeneficii;

```

t_beneficii_imbricat beneficii_imbricat := beneficii_imbricat();

-- VARRAY pentru a memora informatiile despre clienti
TYPE clienti_varray IS VARRAY(100) OF client_info;
t_clienti clienti_varray := clienti_varray();
BEGIN
    BEGIN
        -- Verificare daca exista oferta de vacanta cu ID-ul dat
        SELECT nume_oferta_vacanta
        INTO nume_oferta
        FROM oferta_vacanta
        WHERE id_oferta_vacanta = in_id_oferta;

        EXCEPTION
        WHEN no_data_found THEN
            dbms_output.put_line('Oferta de vacanta cu ID-ul ' || in_id_oferta || ' nu exista.');
```

RETURN;

```

    END;

---memorare informatii despre beneficiile implicate in oferta de vacanta in tabelul indexat
BEGIN
    SELECT ob.id_beneficiu, b.nume_beneficiu BULK COLLECT
    INTO t_beneficii_index
    FROM oferta_implica_beneficiu ob
        JOIN beneficiu b ON ob.id_beneficiu = b.id_beneficiu
    WHERE ob.id_oferta_vacanta = in_id_oferta;

    ---memorare in tabelul imbricat a beneficiilor care nu sunt implicate in oferta de vacanta

```

```

FOR i IN (SELECT b.id_beneficiu, b.ume_beneficiu
          FROM beneficiu b
          WHERE b.id_beneficiu NOT IN (SELECT ob1.id_beneficiu
                                       FROM oferta_implica_beneficiu ob1
                                       WHERE ob1.id_oferta_vacanta = in_id_oferta))
LOOP
    t_beneficii_imbricat.extend;
    t_beneficii_imbricat(t_beneficii_imbricat.LAST).beneficiu_id := i.id_beneficiu;
    t_beneficii_imbricat(t_beneficii_imbricat.LAST).ume := i.ume_beneficiu;
END LOOP;

dbms_output.put_line('-----');
dbms_output.put_line(ume_oferta || ':');
dbms_output.put_line('-----');

IF t_beneficii_index.COUNT = 0 THEN

    dbms_output.put_line('Nu exista beneficii pentru oferta de vacanta cu ID-ul ' ||
in_id_oferta);
    dbms_output.put_line('-----');
ELSE
    dbms_output.put_line('Beneficii implicate:');
    FOR i IN t_beneficii_index.FIRST..t_beneficii_index.LAST
    LOOP
        dbms_output.put_line(' ' || t_beneficii_index(i).ume);
    END LOOP;
END IF;

IF t_beneficii_imbricat.COUNT = 0 THEN

```

```

        dbms_output.put_line('TOATE BENEFICIILE SUNT IMPLICATE');
        dbms_output.put_line('-----');
ELSE
        dbms_output.put_line('-----');
        dbms_output.put_line('Beneficii care nu sunt implicate:');

        FOR i IN t_beneficii_imbricat.FIRST..t_beneficii_imbricat.LAST
        LOOP
                dbms_output.put_line(' ' || t_beneficii_imbricat(i).nume);
        END LOOP;

END IF;

--Memorarea clientilor care au achizitionat oferta de vacanta
SELECT c.id_cumparator, c.num_cumparator, c.prenume_cumparator, c.numar_telefon,
c.data_achizitie
BULK COLLECT INTO t_clienti
FROM oferta_implica_cumparator oc
        JOIN cumparator c ON oc.id_cumparator = c.id_cumparator
WHERE oc.id_oferta_vacanta = in_id_oferta;

dbms_output.put_line('-----');
IF t_clienti.COUNT = 0 THEN
        dbms_output.put_line('Nu exista clienti care au achizitionat oferta. ');
ELSE
        dbms_output.put_line('Informatii despre clientii care au cumparat oferta: ');
        FOR i IN 1..t_clienti.COUNT
        LOOP
                dbms_output.put_line('ID: ' || t_clienti(i).id_cumparator ||
                        ', Nume: ' || t_clienti(i).num_cumparator ||

```

```

        ' || t_clienti(i).prenume_cumparator ||
        ', Telefon: ' || t_clienti(i).numar_telefon ||
        ', Data Achizitie: ' || t_clienti(i).data_achizitie);

    END LOOP;

    END IF;

    dbms_output.put_line('-----');

    END;

END Cerinta6;

PROCEDURE Cerinta7(
    in_oferta_id IN OFERTA_VACANTA.id_oferta_vacanta%TYPE
)
IS
    -- Declaram un cursor implicit pentru informatiile despre oferta de vacanta
    CURSOR oferta_cursor IS
        SELECT id_oferta_vacanta, nume_oferta_vacanta, id_agentie_turism
        FROM oferta_vacanta
        WHERE id_oferta_vacanta < in_oferta_id;

    -- Declaram un cursor parametrizat dependent de primul cursor
    CURSOR agentie_cursor(p_cursor_oferta IN oferta_cursor%ROWTYPE) IS
        SELECT id_agentie_turism, nume_agentie, adresa
        FROM agentie_turism
        WHERE id_agentie_turism = p_cursor_oferta.id_agentie_turism;

    v_oferta_rec oferta_cursor%ROWTYPE;
    v_agentie_rec agentie_cursor%ROWTYPE;

```

```
BEGIN
```

```
OPEN oferta_cursor;
```

```
LOOP
```

```
    FETCH oferta_cursor INTO v_oferta_rec;
```

```
    EXIT WHEN oferta_cursor%NOTFOUND;
```

```
        DBMS_OUTPUT.PUT_LINE('Oferta ID: ' || v_oferta_rec.id_oferta_vacanta || ', Nume: ' ||  
v_oferta_rec.nume_oferta_vacanta);
```

```
-- Deschidem al doilea cursor, transmitem rezultatele primului ca parametru
```

```
OPEN agentie_cursor(v_oferta_rec);
```

```
FETCH agentie_cursor INTO v_agentie_rec;
```

```
IF agentie_cursor%FOUND THEN
```

```
    DBMS_OUTPUT.PUT_LINE(' Agentie ID: ' || v_agentie_rec.id_agentie_turism || ',  
Nume: ' || v_agentie_rec.nume_agentie || ', Adresa: ' || v_agentie_rec.adresa);
```

```
END IF;
```

```
CLOSE agentie_cursor;
```

```
DBMS_OUTPUT.PUT_LINE('-----');
```

```
END LOOP;
```

```
CLOSE oferta_cursor;
```

```
END Cerinta7;
```

```
FUNCTION Cerinta8(in_id_cumparator IN CUMPARATOR.id_cumparator%TYPE)
RETURN NUMBER
```

```
IS
```

```
    v_numar_vacante NUMBER;
```

```
    NoDataFoundException EXCEPTION;
```

```
    InvalidInputException EXCEPTION;
```

```
BEGIN
```

```
    -- Verificam daca parametrul este un numar intreg pozitiv
```

```
    IF in_id_cumparator IS NULL OR in_id_cumparator <= 0 THEN
```

```
        RAISE InvalidInputException;
```

```
    END IF;
```

```
    v_numar_vacante := 0;
```

```
    SELECT COUNT(DISTINCT oic.id_oferta_vacanta)
```

```
    INTO v_numar_vacante
```

```
    FROM OFERTA_IMPLICA_CUMPARATOR oic
```

```
    JOIN OFERTA_IMPLICA_BENEFICIU oib ON oic.id_oferta_vacanta =
oib.id_oferta_vacanta
```

```
    WHERE oic.id_cumparator = in_id_cumparator;
```

```
    IF v_numar_vacante > 0 THEN
```

```
        RETURN v_numar_vacante;
```

```
    ELSE
```

```
        RAISE NoDataFoundException;
```

```
    END IF;
```

```
EXCEPTION
```

```
    WHEN NoDataFoundException THEN
```

```

        DBMS_OUTPUT.PUT_LINE('Nu s-au gasit date pentru cumparatorul cu ID-ul: ' ||
in_id_cumparator);
        RETURN -1;
    WHEN InvalidInputException THEN
        DBMS_OUTPUT.PUT_LINE('Parametrul trebuie sa fie un numar intreg pozitiv.');
```

```

        RETURN -1;
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('A aparut o eroare neasteptata.');
```

```

        RETURN -1;
END Cerinta8;
```

```

PROCEDURE Cerinta9(p_nume_tara VARCHAR2) AS
tara_not_found EXCEPTION;
```

```

BEGIN
    DECLARE
        v_count NUMBER;
    BEGIN
        -- Verifica daca exista tara în tabelul de tari
        SELECT COUNT(*)
            INTO v_count
            FROM TARA t
            WHERE UPPER(t.nume_tara) = UPPER(p_nume_tara);

        IF v_count = 0 THEN
            RAISE tara_not_found;
        END IF;

        -- Verifică dacă există date pentru țara specificată
```



```

SELECT COUNT(DISTINCT mt.denumire_transport)
    INTO v_count
FROM TARA t
JOIN ORAS o ON t.id_tara = o.id_tara
JOIN CAZARE c ON o.id_oras = c.id_oras
JOIN OFERTA_VACANTA ov ON t.id_tara = ov.id_tara
JOIN OFERTA_IMPLICA_TRANSPORT oit ON ov.id_oferta_vacanta =
oit.id_oferta_vacanta
JOIN MOD_TRANSPORT mt ON oit.id_mod_transport = mt.id_mod_transport
WHERE UPPER(t.nume_tara) = UPPER(p_nume_tara);

IF v_count = 0 THEN
    RAISE NO_DATA_FOUND;
ELSIF v_count > 2 THEN
+
    RAISE TOO_MANY_ROWS;
END IF;

-- Afisează rezultatele
FOR modalitate IN (SELECT c.nume_cazare,
                        o.nume_oras,
                        t.nume_tara,
                        LISTAGG(mt.denumire_transport, ' sau ') WITHIN GROUP (ORDER
BY mt.denumire_transport) AS modalitati_transport
                    FROM TARA t
                    JOIN ORAS o ON t.id_tara = o.id_tara
                    JOIN CAZARE c ON o.id_oras = c.id_oras
                    JOIN OFERTA_VACANTA ov ON t.id_tara = ov.id_tara

```

```

        JOIN OFERTA_IMPLICA_TRANSPORT oit ON ov.id_oferta_vacanta =
oit.id_oferta_vacanta

        JOIN MOD_TRANSPORT mt ON oit.id_mod_transport = mt.id_mod_transport

        WHERE UPPER(t.numa_tara) = UPPER(p_numa_tara)

        GROUP BY c.numa_cazare, o.numa_oras, t.numa_tara)

LOOP

    DBMS_OUTPUT.PUT_LINE('Pentru a ajunge la ' || modalitate.numa_cazare || ', ' ||

        modalitate.numa_oras || ', ' || modalitate.numa_tara ||

        ', vei folosi mijloacele de transport: ' || modalitate.modalitati_transport);

END LOOP;

EXCEPTION

    WHEN tara_not_found THEN

        DBMS_OUTPUT.PUT_LINE('Nu exista oferte de vacanta cu destinatia: ' || p_numa_tara);

    WHEN NO_DATA_FOUND THEN

        DBMS_OUTPUT.PUT_LINE('Nu exista modalitati de transport incluse in ofertele de

vacanta din tara: ' || p_numa_tara);

    WHEN TOO_MANY_ROWS THEN

        DBMS_OUTPUT.PUT_LINE('Prea multe date pentru tara specificata.');
```

```

    WHEN OTHERS THEN

        DBMS_OUTPUT.PUT_LINE('A aparut o eroare neasteptata.');
```

```

END;

END Cerinta9;

END cerinta13;

/
```

The screenshot shows the Oracle SQL Developer interface. The top pane displays a PL/SQL script with line numbers 246 to 263. The script includes a GROUP BY clause, a LOOP, and an EXCEPTION block. The bottom pane shows the execution results, indicating that the package CERINTA13 was compiled successfully.

```

246      GROUP BY c.numa_cazare, o.numa_oras, t.numa_tara)
247
248      LOOP
249          DBMS_OUTPUT.PUT_LINE('Pentru a ajunge la ' || modalitate.numa_cazare || ', ' ||
250              modalitate.numa_oras || ', ' || modalitate.numa_tara ||
251              ', vei folosi mijloacele de transport: ' || modalitate.modalitati_transport);
252
253      END LOOP;
254
255      EXCEPTION
256
257      WHEN tara_not_found THEN
258          DBMS_OUTPUT.PUT_LINE('Nu exista oferte de vacanta cu destinatia: ' || p_numa_tara);
259
260      WHEN NO_DATA_FOUND THEN
261          DBMS_OUTPUT.PUT_LINE('Nu exista modalitati de transport incluse in ofertele de vacanta din tara: ' || p_numa_tara);
262
263      WHEN TOO_MANY_ROWS THEN
264          DBMS_OUTPUT.PUT_LINE('Prea multe date pentru tara specificata.');
```

```

265      WHEN OTHERS THEN
266          DBMS_OUTPUT.PUT_LINE('A aparut o eroare neasteptata.');
```

```

267
268      END;
269
270      END Cerinta9;
271
272      END cerinta13;
273
274      /
```

Package CERINTA13 compiled

Package Body CERINTA13 compiled